

Data-Aided Fractional Delay-Doppler Channel Estimation with Embedded Pilot Frames in DZT-Based OTFS

Sai Pradeep Muppaneni, Sandesh Rao Mattu, and A. Chockalingam
Department of ECE, Indian Institute of Science, Bangalore

Abstract—Orthogonal time frequency space (OTFS) implementation using discrete Zak transform (DZT) has performance and complexity advantages in channels with high Doppler spreads. In this paper, we propose a data-aided approach to channel estimation in DZT-based OTFS systems with fractional delays and Dopplers. The proposed method employs an embedded pilot frame (consisting of pilot and data symbols) for channel estimation and leverages detected data to enhance channel estimation accuracy. The algorithm adopts a path-by-path approach where the channel parameters for one path are estimated before proceeding to the next, without assuming the knowledge of the number of paths. Considering the embedded nature of the pilot and data symbols in a frame, the effect of data signal-to-noise ratio (SNR) on the channel estimation performance and the effect of pilot SNR on the data detection performance are evaluated, leading to the data-aided approach to achieve improved performance. Simulation results show that the proposed estimation algorithm achieves significantly better performance compared to threshold based estimation algorithm.

Index Terms—OTFS modulation, delay-Doppler domain, discrete Zak transform, fractional delay-Doppler, data-aided channel estimation.

I. INTRODUCTION

Orthogonal time frequency space (OTFS) modulation has emerged as a promising modulation scheme that offers good resilience to high Doppler spreads of the channel [1]-[5]. A key attribute of OTFS modulation is that it operates by encoding information symbols in the delay-Doppler (DD) domain before transforming them into the time domain (TD) for transmission. In the past, several works have carried out this transformation using a multicarrier (MC) modulation block preceded by a two-dimensional precoding operation using inverse symplectic finite Fourier transform [6]. This scheme is referred to as the multicarrier OTFS (MC-OTFS) scheme [9],[10]. In the MC-OTFS scheme, the transmitter first converts the information symbols from DD domain to time-frequency (TF) domain, and then converts the TF domain output to time domain (TD) for transmission. At the receiving end, the received TD signal undergoes conversion to TF domain before being reverted back to DD domain for signal detection.

An alternate approach to implement OTFS is through the use of Zak transform [7]. In this approach, the DD domain symbols are directly transformed to TD using inverse Zak transform at the transmitter, and the TD signal is transformed back to DD domain at the receiver using Zak transform. A comparative performance analysis between MC-OTFS and Zak-OTFS has revealed superior performance of Zak-OTFS [9],[10]. Drawing inspiration from the FFT-based implementation of OFDM, a

discrete Zak transform (DZT) based implementation emerges as a method to realize Zak transform based OTFS in the discrete domain [11],[12]. The DZT of a sequence is analogous to the discrete Fourier transform of a sub-sampled sequence [11], and therefore possesses computational efficiency. The input-output relation governing a DZT-based OTFS system is derived in [12], and its bit error performance is investigated in [13]. The bit error performance reported in [13] substantiates that DZT-OTFS is more robust to large Doppler spreads than MC-OTFS. A low-complexity maximal ratio combining detector for DZT-OTFS is proposed in [14]. The above works on DZT-OTFS have assumed perfect channel knowledge. In this paper, we focus on DD channel estimation for DZT-OTFS and its performance.

In the pursuit of DD channel estimation at the OTFS receiver, diverse methodologies have been introduced in the literature [5],[16]-[19]. Two key issues of interest in this regard are: *i*) placement of pilot symbol(s) in an OTFS frame, and *ii*) fractional nature of the delays and Dopplers. A simple pilot placement strategy is to have an exclusive pilot frame for channel estimation, which is inefficient due to increased overhead [15],[16]. Another strategy is to use an embedded pilot approach where both pilot and data symbols are placed in a frame with guard symbols in between [17],[18],[19]. An advantage of embedded pilot frames is better spectral efficiency compared to exclusive pilot frames. However, the embedded pilot approach has the associated issue of interference/leakage between pilot and data symbols in the frame. In particular, this issue gets exacerbated in the presence of fractional delays and Dopplers (DDs). This point is illustrated in Fig. 1, which shows the DD channel response for MC-OTFS and DZT-OTFS with integer and fractional DDs. The following two observations in Fig. 1 form the motivation for the work reported in this paper. First, it can be seen that with integer DDs, the channel response is well localized for both MC-OTFS and DZT-OTFS. However, with fractional DDs, the channel response gets diffused across multiple DD bins in the frame. This diffusion is severe in the case of MC-OTFS compared to that in DZT-OTFS. This indicates that DZT-OTFS is a better waveform than MC-OTFS for fractional DDs (which are more practical). Second, although the fractional DD induced diffusion is less in DZT-OTFS, it can still affect channel estimation performance. Simple techniques (e.g., threshold based channel estimation) may prove to be inadequate and algorithms that can suppress the fractional DD induced diffusion effects are needed.

Based on the above, in this paper, we consider the problem of DD channel estimation for DZT OTFS with embedded pilot

This work was supported in part by the J. C. Bose National Fellowship, Department of Science and Technology, Government of India.

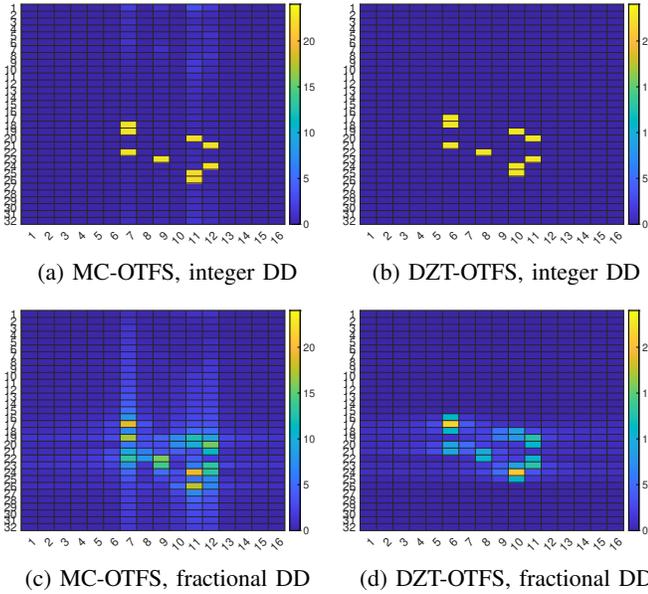


Fig. 1: Channel response for MC-OTFS and DZT-OTFS with 32 delay bins, 16 Doppler bins, and 9 paths.

frames (motivated by better spectral efficiency) in the presence of fractional DDs (which are more practical). Specifically, we propose a channel estimation algorithm where the effect of leakage from data symbols on channel estimation is suppressed by leveraging the detected data in an iterative manner. The algorithm adopts a path-by-path approach where the channel parameters for one path are estimated before proceeding to the next, without assuming the knowledge of the number of paths. Simulation results show that the proposed estimation algorithm achieves significantly better performance compared to threshold based estimation algorithm.

The rest of the paper is organized as follows. The DZT-OTFS system model is presented in Sec. II. The proposed channel estimation algorithm is presented in Sec. III. Simulation results and discussions are presented in Sec. IV. Conclusions and future work are presented in Sec. V.

II. DZT-OTFS SYSTEM MODEL

Consider the matrix $\mathbf{Z}_x \in \mathbb{A}^{L \times K}$ in the DD domain, containing information symbols to be transmitted. Here, L represents the number of bins along the delay domain, K represents the number of bins along the Doppler domain, and \mathbb{A} represents the modulation alphabet. Let W denote the available communication bandwidth, with $W = L\Delta f$, where Δf is the subcarrier spacing. The total time required for transmitting the frame is $\frac{K}{\Delta f}$. The information symbols are positioned on the DD grid at coordinates $(\frac{l}{L\Delta f}, \frac{k\Delta f}{K})$, where $l = 0, \dots, L-1$ and $k = 0, \dots, K-1$. The DD domain information symbols are transformed into a TD vector, denoted by $\mathbf{x} \in \mathbb{C}^{LK \times 1}$, using inverse discrete Zak transform (IDZT). The relationship between \mathbf{x} and \mathbf{Z}_x is given by

$$\mathbf{x} = \text{vec}(\mathbf{Z}_x \mathbf{F}_K^H), \quad (1)$$

where \mathbf{F}_K represents the K -point unitary discrete Fourier transform (DFT) matrix. To prevent inter-frame interference, a cyclic prefix (CP) of length T_{CP} is added to \mathbf{x} , resulting in a vector \mathbf{s} . This process is defined as follows:

$$\mathbf{s}[c] = \begin{cases} \mathbf{x}[(c)_{LK}], & -T_{CP} \leq c \leq LK - 1 \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

To facilitate signal transmission, the vector \mathbf{s} is converted to a continuous-time signal $s(t)$ using the pulse $f(t)$. The continuous-time signal is given by

$$s(t) = \sum_{c=-T_{CP}}^{LK-1} \mathbf{s}[c] f(t - cT_s), \quad (3)$$

where $T_s = 1/W$ is the basic signaling interval. The signal $s(t)$, after passing through the time-varying channel with DD domain impulse response $h(\tau, \nu) = \sum_{i=1}^I \beta_i \delta(\tau - \tau_i) \delta(\nu - \nu_i)$, is observed at the receiver, where I is the number of paths, τ_i, ν_i are the fractional delay and Doppler of the i th path, and β_i is the fade coefficient of the i th path. The received signal, denoted by $r(t)$, at the receiver is given by

$$r(t) = \sum_{i=1}^I \beta_i s(t - \tau_i) e^{j2\pi\nu_i t} + w(t), \quad (4)$$

where $w(t)$ is the additive noise. The received signal $r(t)$ is then passed through a matched filter, and the output $y(t)$ is given by

$$y(t) = \int_{-\infty}^{\infty} r(\tau) f^*(\tau - t) d\tau. \quad (5)$$

By substituting equations (3) and (4) into equation (5), we obtain

$$y(t) = \sum_{i=1}^I \beta_i \sum_{c=-T_{CP}}^{LK-1} \mathbf{s}[c] \int_{-\infty}^{\infty} f(\tau - cT_s - \tau_i) f^*(\tau - t) e^{j2\pi\nu_i \tau} d\tau + \tilde{w}(t), \quad (6)$$

where $\tilde{w}(t)$ represents the match-filtered noise. Under the assumption that the maximum Doppler, $\max\{\nu_i\}$, is much smaller than the pulse bandwidth, and denoting $g(t) = \int f(\tau) f^*(\tau - t) d\tau$, $y(t)$ can be approximated as

$$y(t) \approx \sum_{i=1}^I \beta_i e^{j2\pi\tau_i\nu_i} \sum_{c=-T_{CP}}^{LK-1} \mathbf{s}[c] e^{j2\pi\nu_i c T_s} g(t - cT_s - \tau_i) + \tilde{w}(t). \quad (7)$$

The signal $y(t)$ is sampled at rate $1/T_s$ to obtain the discrete signal vector \mathbf{y} as

$$\mathbf{y}[u] = \sum_{i=1}^I \beta_i e^{j2\pi\tau_i\nu_i} \sum_{c=-T_{CP}}^{LK-1} \mathbf{s}[c] e^{j2\pi c\nu_i T_s} g_i[u - c] + \tilde{w}[u], \quad (8)$$

where $\tilde{w}[u] = \tilde{w}(uT_s)$, $g_i(u) = g(uT_s - \tau_i)$ is assumed to have finite support such that the range of the support is much

less than LK . Now, by defining $\tilde{g}_i[c]$ as the periodic version of $g_i[c]$ with the period LK , (8) can be approximated as

$$y[u] \approx \sum_{i=1}^I \beta_i e^{j2\pi \frac{l_{\tau_i} k_{\nu_i}}{LK}} \sum_{c=0}^{LK-1} s[c] e^{j2\pi c \frac{k_{\nu_i}}{LK}} \tilde{g}_i[u-c] + \tilde{w}[u], \quad (9)$$

after removing the CP. In the above equation, $k_{\nu_i} = \nu_i LKT_s \in \mathbb{R}$, and $l_{\tau_i} = \frac{\tau_i}{T_s} \in \mathbb{R}^+$, i.e., the delays and Dopplers are fractional. Equation (9) can be written in a vector form as

$$\mathbf{y} = \sum_{i=1}^I \beta_i e^{j2\pi \frac{l_{\tau_i} k_{\nu_i}}{LK}} [(\mathbf{x} \cdot \mathbf{v}_i) \otimes \tilde{\mathbf{g}}_i] + \tilde{\mathbf{w}}, \quad (10)$$

where $\mathbf{v}_i[u] = e^{j2\pi u \frac{k_{\nu_i}}{LK}}$, $\mathbf{x} \cdot \mathbf{v}_i$ denotes the element-wise product of \mathbf{x} and \mathbf{v}_i , \otimes is the circular convolution operator, and $\tilde{\mathbf{w}}$ is the additive noise vector. The vector \mathbf{y} is transformed back to the DD domain using DZT to obtain \mathbf{Z}_y as

$$\mathbf{Z}_y[l, k] = \frac{1}{\sqrt{K}} \sum_{n=0}^{K-1} \mathbf{y}[l+nL] e^{-j2\pi \frac{nk}{K}}, \quad (11)$$

where $l = 0, \dots, L-1$ and $k = 0, \dots, K-1$. Using the properties of DZT, equation (11) can be written as

$$\mathbf{Z}_y = \sum_{i=1}^I \beta_i e^{j2\pi \tau_i \nu_i} \mathbf{Z}_{y_i} + \mathbf{Z}_{\tilde{w}}, \quad (12)$$

where

$$\mathbf{Z}_{y_i}[l, k] = \sum_{m=0}^{L-1} \left(\sum_{n=0}^{K-1} \mathbf{z}_x[m, n] \mathbf{z}_{v_i}[m, k-n] \right) \mathbf{z}_{\tilde{g}_i}[l-m, k], \quad (13)$$

\mathbf{Z}_x , \mathbf{Z}_{v_i} , $\mathbf{Z}_{\tilde{g}_i}$, and $\mathbf{Z}_{\tilde{w}}$ are the Zak transforms of \mathbf{x} , \mathbf{v}_i , $\tilde{\mathbf{g}}_i$, and $\tilde{\mathbf{w}}$, respectively.

A. Vectorization of input-output relation

Let \mathbf{z}_y , \mathbf{z}_{y_i} , \mathbf{z}_x , and $\mathbf{z}_{\tilde{w}}$ represent the vectorized forms of matrices \mathbf{Z}_y , \mathbf{Z}_{y_i} , \mathbf{Z}_x , and $\mathbf{Z}_{\tilde{w}}$, respectively. In other words, the $(kL+l)$ th element in the vector corresponds to the $[l, k]$ th entry in the corresponding matrix, such that $\mathbf{Z}_y[l, k] = \mathbf{z}_y[kL+l]$. The vectorized form of input-output relation between \mathbf{z}_{y_i} and \mathbf{z}_x is derived as follows.

Let $\mathbf{H} \in \mathbb{C}^{L \times K}$ and $\mathbf{G} \in \mathbb{C}^{2L-1 \times K}$ be two matrices with entries $\mathbf{H}[l, k] = \mathbf{z}_{v_i}[l, k]$ and $\mathbf{G}[l, k] = \mathbf{z}_{\tilde{g}_i}[l-(L-1), k]$, where $l = 0, \dots, L-1$ and $k = 0, \dots, K-1$. Additionally, let $\mathbf{R}_K \in \mathbb{C}^{K \times K}$ be a reversal matrix and \mathbf{P}_K be a basic circulant permutation matrix of size K [20]. Define a matrix $\mathbf{A}_q^{(i)'} \in \mathbb{C}^{L \times K}$ as

$$\mathbf{A}_q^{(i)'}[l, k] = \begin{cases} \mathbf{H}[l, k], & \text{if } l = [q]_L \\ 0, & \text{otherwise,} \end{cases} \quad (14)$$

where $[\cdot]_L$ denotes the modulo- L operation and $q = 0, \dots, LK-1$. Let $\mathbf{A}_1^{(i)} \in \mathbb{C}^{LK \times LK}$ be a matrix whose q th row is filled with $\text{vec}(\mathbf{A}_q^{(i)'} \mathbf{R}_K \mathbf{P}_K^{\lfloor \frac{q}{L} \rfloor + 1})$, where $\lfloor \cdot \rfloor$ denotes the floor operator. Define $\mathbf{A}_q^{(i)''} \in \mathbb{C}^{L \times K}$ as

$$\mathbf{A}_q^{(i)''}[l, k] = \begin{cases} \mathbf{G}[l + [q]_L, k], & \text{if } k = \lfloor \frac{q}{L} \rfloor \\ 0, & \text{otherwise.} \end{cases} \quad (15)$$

Additionally, define $\mathbf{A}_2^{(i)} \in \mathbb{C}^{LK \times LK}$ whose q th row is filled with $\text{vec}(\mathbf{R}_L \mathbf{A}_q^{(i)'})$. Finally, (13) and (12) can be vectorized as

$$\mathbf{z}_{y_i} = \mathbf{A}^{(i)} \mathbf{z}_x \quad (16)$$

and

$$\mathbf{z}_y = \sum_{i=1}^I \beta_i e^{j2\pi \frac{l_{\tau_i} k_{\nu_i}}{LK}} \mathbf{z}_{y_i} + \mathbf{z}_{\tilde{w}}, \quad (17)$$

respectively, where $\mathbf{A}^{(i)} = \mathbf{A}_2^{(i)} \mathbf{A}_1^{(i)}$. Here, the matrix $\mathbf{A}_1^{(i)}$ effectively performs element-wise multiplication with \mathbf{v}_i , and $\mathbf{A}_2^{(i)}$ carries out the circular convolution with $\tilde{\mathbf{g}}_i$ in (10).

III. PROPOSED CHANNEL ESTIMATION ALGORITHM

In this section, we first present an algorithm to obtain an initial estimate of the channel parameters, and then enhance the estimation accuracy using the knowledge of detected data in the embedded frame. Consider an embedded frame where the data, pilot, and guard symbols coexist in the frame [17]. The structure of the embedded frame is expressed as

$$\mathbf{z}_x[l, k] = \begin{cases} x_p & l = \frac{L}{2}, k = \frac{K}{2} \\ 0 & (l, k) \in \mathbb{L} \otimes \mathbb{K} \\ x_{lk} \in \mathbb{A} & \text{otherwise,} \end{cases} \quad (18)$$

where x_p and x_{lk} denote the pilot and data symbols, respectively, and \otimes denotes the Cartesian product of two sets. That is, the transmit vector \mathbf{z}_x in (16) consists of pilot, guard (zero), and data symbols as per (18). The sets $\mathbb{L} = \{\frac{L}{2} - \lceil l_{\tau_{\max}} \rceil, \dots, \frac{L}{2} + \lceil l_{\tau_{\max}} \rceil\}$ and $\mathbb{K} = \{\frac{K}{2} - 2\lceil k_{\nu_{\max}} \rceil, \dots, \frac{K}{2} + 2\lceil k_{\nu_{\max}} \rceil\}$, where $l_{\tau_{\max}} = \max_i \{l_{\tau_i}\}$ and $k_{\nu_{\max}} = \max_i \{k_{\nu_i}\}$, define the guard space which is the region around the pilot symbol that is filled with zeros to prevent interference between pilot and data symbols¹. This embedded frame is transmitted through the channel and the received signal \mathbf{z}_y in (17) is processed for estimating the channel and detecting the data.

A. Initial channel estimation

The channel parameters that are to be estimated are $(\mathbf{l}_\tau, \mathbf{k}_\nu, \boldsymbol{\beta})$, where $\mathbf{l}_\tau = [l_{\tau_1} \dots l_{\tau_I}]^T$, $\mathbf{k}_\nu = [k_{\nu_1} \dots k_{\nu_I}]^T$ and $\boldsymbol{\beta} = [\beta_1 \dots \beta_I]^T$. In order to obtain an initial estimate of the above parameters, we use the cost function $\|\mathbf{z}_y - \mathbf{B}(\mathbf{l}_\tau, \mathbf{k}_\nu) \boldsymbol{\beta}\|^2$ and obtain the initial estimate as

$$\hat{\mathbf{l}}_\tau, \hat{\mathbf{k}}_\nu, \hat{\boldsymbol{\beta}} = \underset{\mathbf{l}_\tau, \mathbf{k}_\nu, \boldsymbol{\beta}}{\text{argmin}} \|\mathbf{z}_y - \mathbf{B}(\mathbf{l}_\tau, \mathbf{k}_\nu) \boldsymbol{\beta}\|^2, \quad (19)$$

where $\mathbf{B} = [\mathbf{b}_1(l_{\tau_1}, k_{\nu_1}) \mathbf{b}_2(l_{\tau_2}, k_{\nu_2}) \dots \mathbf{b}_I(l_{\tau_I}, k_{\nu_I})] \in \mathbb{C}^{LK \times I}$, $\mathbf{b}_i = e^{j2\pi \frac{l_{\tau_i} k_{\nu_i}}{LK}} \mathbf{A}^{(i)} \mathbf{z}'_x \in \mathbb{C}^{LK \times 1}$, and \mathbf{z}'_x is obtained by making $x_{lk} = 0$ in (18). The above cost function is inspired by the cost function in [16] used for a pilot-only frame. Here, it is used as an approximate cost function for obtaining the initial estimate of the parameters, which are subsequently refined using the estimated data in the embedded frame.

¹It is important to note that, with integer DD, the guard space avoids the interference between pilot and data symbols. However, with fractional DD, interference between pilot and data arises due to symbol spreading in the DD domain.

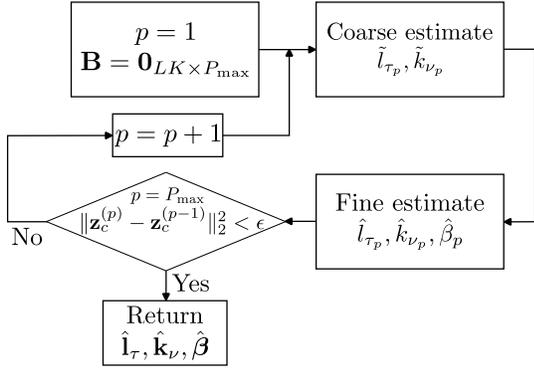


Fig. 2: Flow chart of the channel estimation algorithm to estimate $(\mathbf{l}_\tau, \mathbf{k}_\nu)$ in (21).

Equation (19) is an optimization problem in three variables. To reduce complexity, we initially estimate $(\mathbf{l}_\tau, \mathbf{k}_\nu)$ and then estimate β , as follows. For a given $(\mathbf{l}_\tau, \mathbf{k}_\nu)$, the β which minimizes the cost function in (19) is obtained by differentiating the cost function with respect to β and equating it to zero, and is obtained as

$$\tilde{\beta}(\mathbf{l}_\tau, \mathbf{k}_\nu) = [\mathbf{B}^H(\mathbf{l}_\tau, \mathbf{k}_\nu)\mathbf{B}(\mathbf{l}_\tau, \mathbf{k}_\nu)]^{-1} \mathbf{B}^H(\mathbf{l}_\tau, \mathbf{k}_\nu)\mathbf{z}_y. \quad (20)$$

Using (20) in (19) and simplifying, an estimate of $(\mathbf{l}_\tau, \mathbf{k}_\nu)$ can be obtained as

$$(\hat{\mathbf{l}}_\tau, \hat{\mathbf{k}}_\nu) = \underset{\mathbf{l}_\tau, \mathbf{k}_\nu}{\operatorname{argmax}} \left[\mathbf{z}_y^H \mathbf{B}(\mathbf{l}_\tau, \mathbf{k}_\nu) (\mathbf{B}^H(\mathbf{l}_\tau, \mathbf{k}_\nu)\mathbf{B}(\mathbf{l}_\tau, \mathbf{k}_\nu))^{-1} \mathbf{B}^H(\mathbf{l}_\tau, \mathbf{k}_\nu)\mathbf{z}_y \right]. \quad (21)$$

Using the estimates $(\hat{\mathbf{l}}_\tau, \hat{\mathbf{k}}_\nu)$ from (21) in (20), an estimate of β is obtained as

$$\hat{\beta}(\hat{\mathbf{l}}_\tau, \hat{\mathbf{k}}_\nu) = [\mathbf{B}^H(\hat{\mathbf{l}}_\tau, \hat{\mathbf{k}}_\nu)\mathbf{B}(\hat{\mathbf{l}}_\tau, \hat{\mathbf{k}}_\nu)]^{-1} \mathbf{B}^H(\hat{\mathbf{l}}_\tau, \hat{\mathbf{k}}_\nu)\mathbf{z}_y. \quad (22)$$

We note that (21) represents the correlation of the received signal with the reconstructed signal at different points in the search area. Solving (21) using an exhaustive search of the entire solution space in fine resolution is computationally expensive. So, we use an algorithm which estimates the channel parameters in two steps, namely, a coarse resolution search to obtain the optimal integer estimates of the parameters, followed by a fine resolution search to obtain the optimal estimates of the fractional parts around the integer estimates. This reduces complexity. Also, the algorithm estimates the channel in a path-by-path manner, where the channel parameters of the p th path are estimated only after all the channel parameters of paths from 1 to $p-1$ are estimated.

Figure 2 shows the flowchart of the estimation algorithm to solve (21). The algorithm begins by initializing the path index $p = 1$ and $\mathbf{B} = \mathbf{0}_{LK \times P_{\max}}$, where P_{\max} is the maximum number of paths that the algorithm estimates. After the initialization, the algorithm starts estimating the parameters of the channel path-by-path. For each path, this estimation involves two stages: *coarse estimation* and *fine estimation*. After the fine estimation stage of each path, the algorithm checks if the stopping criterion is met. If it is not met, p

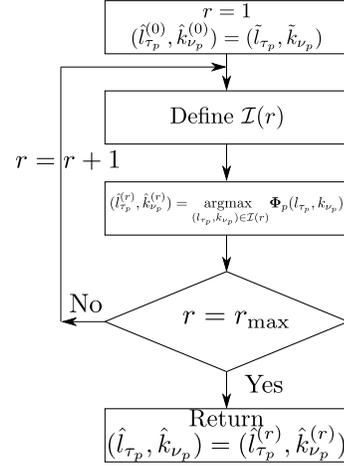


Fig. 3: Flow chart of fine estimation algorithm.

is incremented by 1, and the algorithm proceeds to estimate the parameters of the $(p+1)$ th path. If it is met, then the algorithm terminates and returns the vectors $(\hat{\mathbf{l}}_\tau, \hat{\mathbf{k}}_\nu, \hat{\beta})$. The stopping criterion and coarse and fine estimation stages used in estimating the parameters of p th $(1 \leq p \leq P_{\max})$ path are described below.

1) *Coarse Estimation*: In this stage, we estimate the integer parts of (l_{τ_p}, k_{ν_p}) . Towards this, we define a search area, \mathcal{J} , which is the Cartesian product of two sets \mathcal{L} and \mathcal{K} , where

$$\mathcal{L} = \{0, \dots, \lceil l_{\tau_{\max}} \rceil\}, \quad \mathcal{K} = \{-\lceil k_{\nu_{\max}} \rceil, \dots, 0, \dots, \lceil k_{\nu_{\max}} \rceil\}.$$

A cost function, $\Phi_p(l_{\tau_p}, k_{\nu_p})$, obtained from (21) as

$$\Phi_p(l_{\tau_p}, k_{\nu_p}) = \mathbf{z}_y^H \mathbf{B}_p(\mathbf{l}_\tau, \mathbf{k}_\nu) (\mathbf{B}_p^H(\mathbf{l}_\tau, \mathbf{k}_\nu)\mathbf{B}_p(\mathbf{l}_\tau, \mathbf{k}_\nu))^{-1} \mathbf{B}_p^H(\mathbf{l}_\tau, \mathbf{k}_\nu)\mathbf{z}_y, \quad (23)$$

where $\mathbf{B}_p(\mathbf{l}_\tau, \mathbf{k}_\nu) = [\mathbf{b}_1(\hat{l}_{\tau_1}, \hat{k}_{\nu_1}) \cdots \mathbf{b}_p(l_{\tau_p}, k_{\nu_p}) \mathbf{0} \cdots \mathbf{0}]$, $(\hat{l}_{\tau_i}, \hat{k}_{\nu_i})$ represents the estimates of (l_{τ_i}, k_{ν_i}) for $i = 1, 2, \dots, p-1$, is maximized over all $(l_{\tau_p}, k_{\nu_p}) \in \mathcal{J}$, i.e.,

$$(\tilde{l}_{\tau_p}, \tilde{k}_{\nu_p}) = \underset{(l_{\tau_p}, k_{\nu_p}) \in \mathcal{J}}{\operatorname{argmax}} \Phi_p(l_{\tau_p}, k_{\nu_p}). \quad (24)$$

2) *Fine Estimation*: After the coarse estimation stage of the p th path, the algorithm proceeds to the this stage, where the fractional parts of the delay and Doppler of the p th path are estimated. This is done iteratively by increasing the resolution of the estimation (using a parameter r) in each iteration as described below.

Figure 3 shows the flow chart for the iterative fine estimation algorithm. This fine estimation algorithm is initialized by setting $r = 1$ and $(\hat{l}_{\tau_p}^{(0)}, \hat{k}_{\nu_p}^{(0)}) = (\tilde{l}_{\tau_p}, \tilde{k}_{\nu_p})$, where $(\hat{l}_{\tau_p}^{(r)}, \hat{k}_{\nu_p}^{(r)})$ denotes the fine estimate of (l_{τ_p}, k_{ν_p}) in r th iteration. In the r th iteration, a search area, $\mathcal{I}(r)$, is defined as

$$\mathcal{I}(r) = \left\{ \left\{ l_{\tau_p}^{(r-1)} - \frac{5}{10^r}, l_{\tau_p}^{(r-1)} - \frac{4}{10^r}, \dots, l_{\tau_p}^{(r-1)} + \frac{5}{10^r} \right\} \otimes \left\{ k_{\nu_p}^{(r-1)} - \frac{5}{10^r}, k_{\nu_p}^{(r-1)} - \frac{4}{10^r}, \dots, k_{\nu_p}^{(r-1)} + \frac{5}{10^r} \right\} \right\}. \quad (25)$$

The cost function in (23) is maximized over $\mathcal{I}(r)$ to obtain $(\hat{l}_{\tau_p}^{(r)}, \hat{k}_{\nu_p}^{(r)})$ as

$$(\hat{l}_{\tau_p}^{(r)}, \hat{k}_{\nu_p}^{(r)}) = \underset{(l_{\tau_p}, k_{\nu_p}) \in \mathcal{I}(r)}{\operatorname{argmax}} \Phi_p(l_{\tau_p}, k_{\nu_p}). \quad (26)$$

Following this, the value of r is incremented by 1 and this algorithm proceeds to the next iteration. This iterative procedure continues until $r = r_{\max}$. At this point, the fine estimation algorithm is terminated with the p th path parameter estimates as $(\hat{l}_{\tau_p}, \hat{k}_{\nu_p}) = (\hat{l}_{\tau_p}^{(r_{\max})}, \hat{k}_{\nu_p}^{(r_{\max})})$.

3) *Stopping Criterion:* After the fine estimation stage of the p th path, the channel estimation algorithm stops if $p = P_{\max}$ or $\|\mathbf{z}_c^{(p)} - \mathbf{z}_c^{(p-1)}\|^2 < \epsilon$, where $\mathbf{z}_c^{(p)} = \mathbf{B}_p(\mathbf{1}_\tau, \mathbf{k}_\nu)(\mathbf{B}_p^H(\mathbf{1}_\tau, \mathbf{k}_\nu)\mathbf{B}_p(\mathbf{1}_\tau, \mathbf{k}_\nu))^{-1}\mathbf{B}_p^H(\mathbf{1}_\tau, \mathbf{k}_\nu)\mathbf{z}_y$ and ϵ is a parameter that determines the number of paths estimated. Once the stopping criterion is met, $\hat{\beta}$ is obtained using (22).

The initial channel estimation algorithm presented above estimates the channel without considering the effect of data. However, due to the fractional nature of the channel, the channel estimation performance is hindered by interference between pilot and data symbols. It is therefore necessary and possible to further improve the accuracy of channel estimation using the aid of detected data. This is referred to as data-aided channel estimation and is presented below.

B. Data-aided channel estimation

The estimate obtained from the initial channel estimation is used to construct the estimated channel matrix, $\hat{\mathbf{A}} = \sum_i \hat{\beta}_i e^{j2\pi \frac{\hat{l}_{\tau_i} \hat{k}_{\nu_i}}{LK}} \hat{\mathbf{A}}^{(i)}$, where $\hat{\mathbf{A}}^{(i)}$ is obtained from $\mathbf{A}^{(i)}$ by replacing $\mathbf{1}_\tau, \mathbf{k}_\nu$ with $\hat{\mathbf{1}}_\tau, \hat{\mathbf{k}}_\nu$. Using $\hat{\mathbf{A}}$, data detection is carried out in the received frame². We use this detected data to re-estimate the channel. Towards this, we define $\hat{\mathbf{z}}_x = \operatorname{vec}(\hat{\mathbf{Z}}_x)$, where

$$\hat{\mathbf{Z}}_x[l, k] = \begin{cases} x_p & l = \frac{L}{2}, k = \frac{K}{2} \\ 0 & (l, k) \in \mathbb{L} \otimes \mathbb{K} \\ \hat{x}_{lk} \in \mathbb{A} & \text{otherwise,} \end{cases} \quad (27)$$

and \hat{x}_{lk} denotes the detected data. The re-estimation of the channel parameters is carried out as shown in Fig. 2 by replacing \mathbf{z}'_x by $\hat{\mathbf{z}}_x$ in the construction of \mathbf{B} (see (19)). The so obtained estimates are referred to as data-aided channel estimates.

Remark on complexity: Since the channel estimation algorithm doesn't assume the knowledge of the number of paths, and the number of estimated paths is not fixed, we present the worst-case complexity where P_{\max} number of paths are estimated. The total number of arithmetic operations required to evaluate the cost function in (23) is $C = 2P_{\max}LK + 2P_{\max}^2LK + P_{\max}^2 + \mathcal{O}(P_{\max}^3) - 1$. The number of times this cost function is computed in coarse and fine estimation stages is $C' = (\lceil l_{\tau_{\max}} \rceil + 1)(2\lceil k_{\nu_{\max}} \rceil + 1)P_{\max}$

²In this paper, we use linear minimum mean square error (LMMSE) equalizer for the detection purpose and use the whole received frame for the detection instead of using data bins alone. This is done so that the effect of pilot can be nullified by the equalizer without having to cancel the effect of pilot separately.

and $C'' = 121r_{\max}P_{\max}$, respectively. Therefore, worst-case computational complexity is given by $2C(C' + C'')$, where the factor of 2 is to account for initial and data-aided channel estimations.

Remark on the effect of pilot and data powers on the performance in an embedded frame: To draw a parallel with the exclusive pilot frame, we consider a scenario where the exclusive pilot frame and a data frame (with no pilots and guard) are transmitted together, under the assumption that the channel does not change for the two frames duration. Channel estimates obtained using the exclusive pilot frame are used to detect the data in the subsequent data frame. For this exclusive pilot setup, increasing the power of the pilot frame and the data frame leads to improved NMSE and better detection, respectively. However, this is not always the case in an embedded frame, where the performance may not consistently get better by increasing the power of pilot and data. This happens because of the interaction (interference) between the pilot and data in fractional DD. When the data power is fixed and the pilot power is increased, the normalized mean square error (NMSE) improves as expected. However, this need not translate to improved detection performance. This is because as the pilot power is increased, the spread of the pilot symbol overpowers the data symbols and this is not completely handled during detection due to inaccuracies in channel estimation³. On the other hand, when the pilot power is fixed and the data power is increased, the channel estimation NMSE deteriorates due to stronger spread from data into pilot. However, if the data power is high enough to handle the deterioration, the NMSE improves with data-aid which results in good detection performance.

IV. RESULTS AND DISCUSSIONS

In this section, we present the performance results of the proposed algorithm in the context of embedded pilot frames. The DZT-OTFS system parameters used for the simulation are $L = 64$, $K = 32$, $\Delta f = 15$ kHz. Square-root raised cosine pulse is used. The number of paths in the channel is taken to be 5 with uniform power delay profile. The first path always has a delay of $\tau_1 = 0$, while the subsequent paths have delays $(\tau_i, i = 2, \dots, 5)$ uniformly distributed in $(2(i-1) - 1, 2(i-1)]\mu s$. Furthermore, the maximum Doppler, ν_{\max} is 1700 Hz, and the Doppler values of all the paths are taken to be $\nu_{\max} \cos(\theta_i)$, where θ_i is uniformly distributed in $[0, 2\pi)$. Additionally, parameters governing the algorithm are $P_{\max} = 15$, $n_{\max} = 2$, and $\epsilon = 20\sigma^2$, where σ^2 is the noise variance. Data symbols are taken from a 16-QAM constellation. LMMSE equalizer is used for data detection. Pilot SNR (PSNR) is defined as $\frac{|x_p|^2}{LK\sigma^2}$ and data SNR (DSNR) is defined as $\frac{\mathbb{E}|x_{lk}|^2}{\sigma^2}$.

³The LMMSE equalizer for the embedded frame is $\boldsymbol{\mu}_x + \mathbf{R}_{xx}\mathbf{A}^H(\mathbf{A}\mathbf{R}_{xx}\mathbf{A}^H + \sigma^2\mathbf{I})^{-1}(\mathbf{z}_y - \mathbf{A}\boldsymbol{\mu}_x)$, where $\boldsymbol{\mu}_x = \mathbf{z}'_x$ and \mathbf{R}_{xx} denote the mean and co-variance matrix of \mathbf{z}_x , respectively, \mathbf{A} denotes the channel matrix, and \mathbf{I} denotes the identity matrix. Because of the inaccuracy in estimation of \mathbf{A} , the term $\mathbf{z}_y - \mathbf{A}\boldsymbol{\mu}_x$ does not completely remove the effect of the pilot signal.

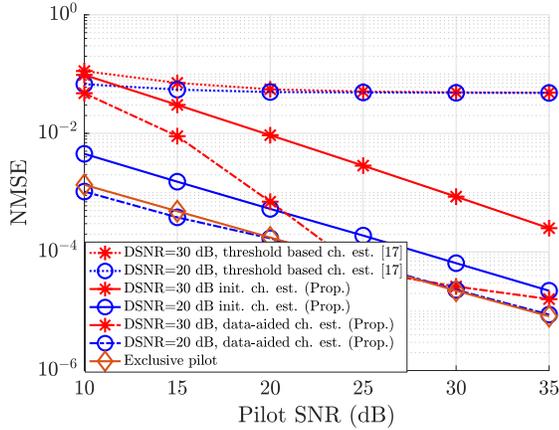


Fig. 4: NMSE performance of the proposed algorithm as a function of pilot SNR.

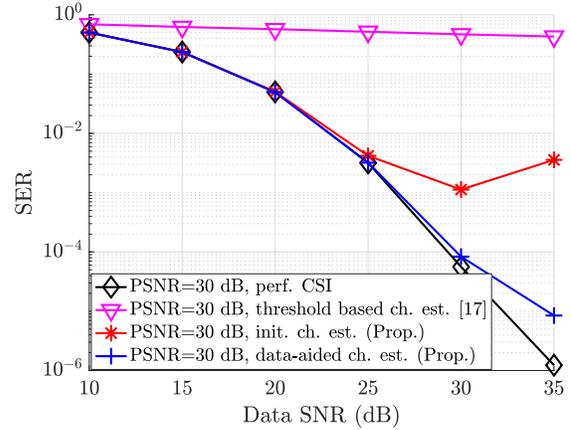


Fig. 6: SER performance of the proposed algorithm as a function of data SNR.

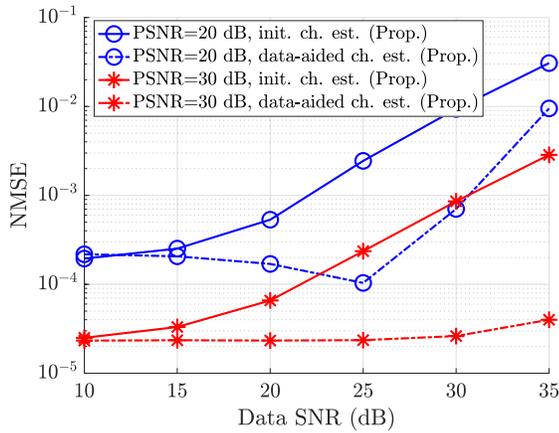


Fig. 5: NMSE performance of the proposed algorithm as a function of data SNR.

Figure 4 shows the performance of the proposed algorithm in terms of NMSE as a function of the PSNR for DSNRs of 20 dB and 30 dB. The NMSE results for the threshold-based estimation scheme in [17] are also provided for comparison. The proposed algorithm shows a linear decrease in NMSE without data aid as the PSNR increases. This is because the algorithm estimates the DD in the fractional domain, as opposed to the integer estimation in the threshold-based scheme even when the channel DD values are fractional. Therefore, the NMSE of the threshold-based scheme floors. With the aid of detected data, the NMSE performance is seen to improve. In some cases, the performance of the proposed algorithm with data aid is observed to match the performance of exclusive pilot. The extent to which this convergence occurs depends on PSNR and DSNR, which determine the accuracy of the initial channel estimate, which, in turn, effects the detected data and the efficacy of data-aided channel estimation.

Figure 5 shows the NMSE performance of the proposed algorithm as a function of DSNR for PSNRs of 20 dB

and 30 dB. It is seen that as DSNR increases, the initial channel estimation NMSE worsens. This is because as the DSNR increases, the interference from the data to the pilot increases and thus increases the error in the channel estimation. However, with data-aid, the NMSE performance is seen to significantly improve. It is also observed that, as PSNR increases, the NMSE decreases, as expected.

Figure 6 shows the symbol error rate (SER) performance of the proposed algorithm as a function of the DSNR for a PSNR of 30 dB. The SER performance of the threshold based algorithm is also presented. Additionally, the SER performance with perfect channel state information (CSI) is also added for comparison. It is seen that the SER of the proposed algorithm decreases with increase in data SNR, whereas the performance of threshold based estimation floors. Also, it is observed that the data-aided channel estimation significantly improves the SER performance compared to that of the initial channel estimation. This is because of the cancellation of the data interference to the pilot which leads to improved channel estimation accuracy. This, in turn, leads to better data detection performance. Further, the SER performance with data-aid closely matches with that achieved using perfect CSI, indicating the effectiveness of the proposed algorithm.

Figure 7 shows the SER performance as a function of PSNR for DSNRs of 20 dB and 30 dB. It is seen that at low PSNRs (< 15 dB), performance of the proposed algorithm (initial and data-aided) with 30 dB DSNR is poorer compared to that with 20 dB DSNR. This is because of the poor channel estimation accuracy at such low PSNRs, as corroborated in Fig. 4. Also, in Fig. 4, it is seen that, at high PSNRs (> 25 dB), the NMSE of data-aided estimation at 30 dB DSNR is similar to that of 20 dB DSNR. While their NMSE performance is similar at high PSNRs, 30 dB DSNR achieves better SER performance compared to 20 dB DSNR, which is due to the higher data power in the case of 30 dB DSNR. Also, it is seen that the proposed initial channel estimation algorithm floors at PSNR of 30 dB, while the data-aided estimation floors at 25 dB. This

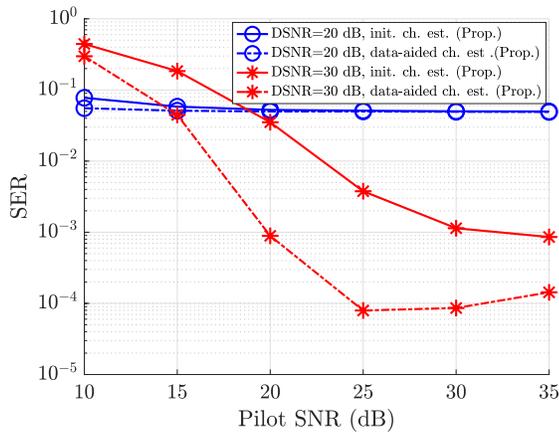


Fig. 7: SER performance of the proposed algorithm as a function of pilot SNR.

is because of the increased interference due to stronger pilot at high PSNRs.

V. CONCLUSIONS

We investigated the problem of fractional DD channel estimation in DZT based OTFS systems with embedded pilot frames. Fractional DD was considered because it is more practical. DZT-OTFS was considered motivated by its better localization of the channel response in fractional DD compared to MC-OTFS. Embedded pilot frames were considered owing to better spectral efficiency. We proposed a channel estimation algorithm which exploited the detected data in the embedded frame to achieve improved channel estimation performance. The proposed algorithm was shown to perform significantly better compared to the threshold based channel estimation. Other channel estimation algorithms for the considered system can be devised and compared as useful future work.

REFERENCES

- [1] R. Hadani et al., "Orthogonal time frequency space modulation," *Proc. IEEE WCNC'2017*, pp. 1-6, Mar. 2017.
- [2] K. R. Murali and A. Chockalingam, "On OTFS modulation for high-Doppler fading channels," *Proc. ITA'2018*, pp. 1-10, Feb. 2018.
- [3] P. Raviteja, K. T. Phan, Y. Hong, and E. Viterbo, "Interference cancellation and iterative detection for orthogonal time frequency space modulation," *IEEE Trans. Wireless Commun.*, vol. 17, no. 10, pp. 6501-6515, Oct. 2018.
- [4] Z. Wei et al., "Orthogonal time-frequency space modulation: a promising next-generation waveform," *IEEE Wireless Commun. Mag.*, vol. 28, no. 4, pp. 136-144, Aug. 2021.
- [5] Y. Hong, T. Thaj, and E. Viterbo, *Delay-Doppler Communications: Principles and Applications*, London UK: Elsevier, 2022.
- [6] "Best readings in orthogonal time frequency space (OTFS) and delay Doppler signal processing," June 2022. Online: <https://www.comsoc.org/ublications/best-readings/orthogonal-time-frequency-space-otfs-and-delay-doppler-signal-processing>.
- [7] A. J. E. M. Janssen, "The Zak transform: a signal transform for sampled time-continuous signals," *Philips J. Res.*, 43, pp. 23-69, 1988.
- [8] S. K. Mohammed, "Derivation of OTFS modulation From first principles," *IEEE Trans. Veh. Tech.*, vol. 70, no. 8, pp. 7619-7636, Aug. 2021.
- [9] S. K. Mohammed, R. Hadani, A. Chockalingam, and R. Calderbank, "OTFS - a mathematical foundation for communication and radar sensing in the delay-Doppler domain," *IEEE BITS the Information Theory Magazine*, vol. 2, no. 2, pp. 36-55, Nov. 2022.

- [10] S. K. Mohammed, R. Hadani, A. Chockalingam, and R. Calderbank, "OTFS - predictability in the delay-Doppler domain and its value to communication and radar sensing," arXiv:2302.08705 [eess.SP] 17 Feb 2023.
- [11] H. Bolcskei and F. Hlawatsch, "Discrete Zak transforms, polyphase transforms, and applications," *IEEE Trans. Signal Process.*, vol. 45, no. 4, pp. 851-866, Apr. 1997.
- [12] F. Lampel, A. Avarado, F. M. J. Willems, "On OTFS using the discrete Zak transform," *IEEE ICC'2022 Workshops*, pp. 729-734, May 2022.
- [13] V. Yogesh, V. S. Bhat, S. R. Mattu, and A. Chockalingam, "On the bit error performance of OTFS modulation using discrete Zak transform," arXiv:2303.12496 [cs.IT] 22 Mar 2023.
- [14] T. Thaj, E. Viterbo, and Y. Hong, "General I/O relations and low-complexity universal MRC detection for all OTFS variants," *IEEE Access*, vol. 10, pp. 96026-96037, 2022.
- [15] M. K. Ramachandran and A. Chockalingam, "MIMO-OTFS in high-Doppler fading channels: signal detection and channel estimation," *Proc. IEEE GLOBECOM'2018*, pp. 206-212, Dec. 2018.
- [16] I. A. Khan and S. K. Mohammed, "Low complexity channel estimation for OTFS modulation with fractional delay and Doppler," arXiv:2111.06009 [cs.IT], Nov. 2021.
- [17] P. Raviteja, K. T. Phan and Y. Hong, "Embedded pilot-aided channel estimation for OTFS in delay-Doppler channels," *IEEE Trans. Veh. Tech.*, vol. 68, no. 5, pp. 4906-4917, May 2019.
- [18] Z. Wei, W. Yuan, S. Li, J. Yuan and D. W. K. Ng, "Off-grid channel estimation with sparse Bayesian learning for OTFS systems," *IEEE Trans. Wireless Commun.*, vol. 21, no. 9, pp. 7407-7426, Sept. 2022.
- [19] I. A. Khan and S. K. Mohammed, "A low complexity OTFS channel estimation method for fractional delay-Doppler scenarios," *IEEE Wireless Commun. Lett.*, early access: doi: 10.1109/LWC.2023.3274936.
- [20] R. Horn and C. Johnson, *Matrix Analysis*, Cambridge Univ. Press, 2013.